REMARKS

I. General

Claims 1-29 were pending in the present application, and all of the pending claims are rejected in the current Office Action (mailed September 6, 2005). The outstanding issues raised in the current Office Action are:

- Claims 1-29 are rejected under 35 U.S.C. § 103(a) as being unpatentable over published European Patent Application EP 0 926 605 to Edwards (hereinafter "*Edwards*") in view of U.S. Patent No. 5,572,673 issued to Shurts (hereinafter "*Shurts*").

In response, Applicant respectfully traverses the outstanding claim rejections, and requests reconsideration and withdrawal thereof in light of the amendments and remarks presented herein.

II. Amendments

Claims 1 and 24-26 are amended and new claims 30-33 are added herein. No new matter is presented by these amendments and claim additions.

Claim 1 is amended herein to specify that the defining arbitrary relationships is in a policy file. It is also amended to specify that the operating system enforces the sensitivity labels except as permitted by the relationships defined in the policy file.

Claim 24 is amended to recite "configuring, in a file, access relationships between entities having different sensitivity labels, wherein said configuring comprises configuring at least one access relationship between entities having incomparable sensitivity labels; and enforcing sensitivity labels by an operating system such that transfer of data between entities associated with incomparable sensitivity labels is restricted except for said at least one access relationship configured in said file." Claim 24 is further amended to delete the recitation of "permitting arbitrary relationships for allowing discrete access between entities of differing sensitivity labels."

Claims 25-26 are amended solely to ensure that proper antecedent basis and clarity are maintained in view of the above amendments to claim 24.

III. Rejections Under 35 U.S.C. § 103(a)

Claims 1-29 are rejected under 35 U.S.C. § 103(a) as being unpatentable over *Edwards* in view of *Shurts*. Applicant respectfully traverses this rejection as discussed below.

To establish a prima facie case of obviousness under 35 U.S.C. § 103(a), three basic criteria must be met. *See* M.P.E.P. § 2143. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the applied reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference must teach or suggest all the claim limitations. Without conceding any other criteria, Applicant respectfully asserts that the applied references do not teach or suggest all the claim limitations, as discussed further below.

Independent Claim 1

Independent claim 1, as amended herein, recites:

> A method for maintaining a secure operating system run-time environment comprising:
> designating sensitivity labels associated with subjects and objects such that each sensitivity label either dominates, is dominated by, or is incomparable to each other sensitivity label;
> defining in a policy file arbitrary relationships between the subjects and objects of differing sensitivity labels; and
> enforcing sensitivity labels by the operating system such that the operating system restricts the transfer of data between subjects and objects associated with inconsistent sensitivity labels except as permitted by said relationships defined in said policy file, thereby providing discrete access between arbitrary, incomparable sensitivity labels. (Emphasis added).

The combination of *Edwards* and *Shurts* fails to teach or suggest all of the above elements of claim 1. For example, the combination of *Edwards* and *Shurts* fails to teach or suggest defining the arbitrary relationships in a policy file, and enforcing the sensitivity labels by an operating system except as permitted by the defined relationships. While *Edwards* teaches the use of MAC sensitivity labels by an operating system to restrict the transfer of data between subjects and objects associated with inconsistent sensitivity labels, it fails to

teach or suggest defining in a policy file arbitrary relationships between the subjects and objects of differing sensitivity labels.

Similarly, *Shurts* mentions the use of MAC sensitivity labels by an operating system to restrict the transfer of data between subjects and objects associated with inconsistent sensitivity labels. Additionally, *Shurts* teaches that a subject can use a certified trusted stored procedure to access objects having sensitivity levels different from the subject's sensitivity level, *see* col. 2, lines 39-56 of *Shurts*. In *Shurts*, a procedure, referred to as a trusted stored procedure, is provided, which may be used by certain subjects to access objects having a sensitivity level that is otherwise inaccessible (according to the MAC rules) by the subject. The trusted stored procedure itself has a sensitivity label associated therewith, which must be compatible with the subject's sensitivity label in order for the subject to utilize the procedure. For instance, *Shurts* explains at column 3, lines 31-56:

> The trusted stored procedures of this invention contain two types of sensitivity labels: (1) read and write sensitivity labels used during execution, and (2) an access sensitivity label used to determine whether a subject can initiate execution of the stored procedure. The second of these, the access sensitivity label, is somewhat analogous to a conventional MAC label associated with an object. If the subject's read sensitivity label dominates the trusted stored procedure's access sensitivity label, the subject is granted access to the procedure. The read and write sensitivity labels used during execution of a trusted stored procedure have no counterpart in standard MAC policies. If a trusted stored procedure's read sensitivity label dominates a database object's access sensitivity label, the trusted stored procedure can read that object during execution. Similarity if a trusted stored procedure's write sensitivity label is dominated by an object's access sensitivity label, the trusted stored procedure can write to that object during execution. A subject's sensitivity labels need not dominate the trusted stored procedure's read and write labels in order for the trusted stored procedure to execute. In fact, a trusted stored procedure may access objects beyond the reach of the subject in normal operation. In preferred embodiments, such access is available only in the controlled environment of the certified trusted stored procedure so that the risk of a security breach is minimized or eliminated.

Also, in *Shurts*, each object has associated therewith a certification state, where the trusted stored procedure does not allow access by a subject to an object that has a sensitivity label inconsistent with that of the requesting subject, unless the certification state indicates that the object is "certified." "Certification indicates that (1) a security officer has evaluated

and certified the object, and (2) a certified object has not undergone a defined security-relevant change since certification." Col. 2, lines 43-46 of *Shurts*.

Thus, *Shurts* provides a trusted stored procedure that can be invoked by a requesting subject (if the requesting subject's sensitivity label is compatible with the trusted stored procedure's sensitivity label), and the trusted stored procedure may allow the requesting subject to access an incompatible object if the incompatible object is certified. However, *Shurts* provides no teaching or suggestion of defining in a policy file arbitrary relationships between the subjects and objects of differing sensitivity labels, and enforcing sensitivity labels by the operating system such that the operating system restricts the transfer of data between subjects and objects associated with inconsistent sensitivity labels except as permitted by said relationships defined in said policy file, as recited by claim 1. *Shurts* provides no hint of a policy file that defines relationships, where the operating system can restrict transfers between subjects and objects associated with inconsistent sensitivity labels except as permitted by the relationships defined in the policy file. Instead, *Shurts* teaches a different approach, wherein a trusted procedure is provided that can be invoked by a requesting subject to enable access to certified objects. The *Shurts* approach requires that the requesting subject utilize a particular trusted procedure in order to access otherwise inaccessible objects, and does suggest defining relationships in a policy file for use by the operating system in determining whether to enforce inconsistent sensitivity labels.

Thus, in view of the above, the combination of *Edwards* and *Shurts* fails to teach or suggest all elements of independent claim 1. Therefore, the rejection of claim 1 should be withdrawn.

Independent Claim 12

Independent claim 12 recites:

> A secure operating system in which sensitivity labels, each comprised of a security level and one or more compartments, are enforced such that the operating system restricts the transfer of data between subjects and objects of differing sensitivity labels where a sensitivity label must dominate or be considered incomparable to other sensitivity labels comprising:
>
> a label encodings file comprising:
>
> a classification section to define the hierarchical names of the system in which classifications are ranked hierarchically according to an assigned level from lower to higher, wherein the higher classifications dominate the lower classifications,
>
> a compartment section to define subdivisions of data possible within a classification,
>
> a label section to define valid labels and their tag values, and
>
> a communications section defining allowed communications channels between privileged and non-privileged processes that do not possess the same sensitivity label, wherein one or more channels are arbitrarily defined. (Emphasis added).

The combination of *Edwards* and *Shurts* fails to teach or suggest all of the above elements of claim 12. For example, the combination of *Edwards* and *Shurts* fails to teach or suggest the recited label encodings file that comprises a communications section defining allowed communications channels between privileged and non-privileged processes that do not possess the same sensitivity label. While *Edwards* teaches the use of MAC sensitivity labels by an operating system to restrict the transfer of data between subjects and objects associated with inconsistent sensitivity labels, it fails to teach or suggest such a label encodings file.

Similarly, *Shurts* mentions the use of MAC sensitivity labels by an operating system to restrict the transfer of data between subjects and objects associated with inconsistent sensitivity labels. However, as discussed above with claim 1, *Shurts* fails to teach or suggest a label encodings file that comprises a communications section defining allowed communications channels between privileged and non-privileged processes that do not possess the same sensitivity label. Instead, *Shurts* teaches a different approach, wherein a trusted procedure is provided that can be invoked by a requesting subject to enable access to certified objects. The *Shurts* approach requires that the requesting subject utilize a particular

trusted procedure in order to access otherwise inaccessible objects, and does suggest defining communications channels in a label encodings file, as recited by claim 12.

Thus, in view of the above, the combination of *Edwards* and *Shurts* fails to teach or suggest all elements of independent claim 12. Therefore, the rejection of claim 12 should be withdrawn.

Independent Claim 24

Independent claim 24, as amended herein, recites:

> A method for providing discrete access control between entities associated with sensitivity labels, each label comprising both a classification level component and a compartment component, comprising:
> defining a fixed set of classifications for each entity;
> defining a dynamic set of compartments for each entity;
> partitioning application process entities and network interface entities into unique compartments;
> configuring, in a file, access relationships between entities having different sensitivity labels, wherein said configuring comprises configuring at least one access relationship between entities having incomparable sensitivity labels; and
> enforcing sensitivity labels by an operating system such that transfer of data between entities associated with incomparable sensitivity labels is restricted except for said at least one access relationship configured in said file. (Emphasis added).

The combination of *Edwards* and *Shurts* fails to teach or suggest all of the above elements of claim 24. For example, the combination of *Edwards* and *Shurts* fails to teach or suggest the recited configuring, in a file, access relationships between entities having different sensitivity labels, and enforcing the sensitivity labels by an operating system except for the relationship(s) configured in the file. While *Edwards* teaches the use of MAC sensitivity labels by an operating system to restrict the transfer of data between subjects and objects associated with inconsistent sensitivity labels, it fails to teach or suggest such a file that is configured to define permitted access relationships between entities having different sensitivity labels.

Similarly, *Shurts* mentions the use of MAC sensitivity labels by an operating system to restrict the transfer of data between subjects and objects associated with inconsistent sensitivity labels. However, as discussed above with claim 1, *Shurts* fails to teach or suggest

a file that is configured to defined permitted access relationships between entities having different sensitivity labels. Instead, *Shurts* teaches a different approach, wherein a trusted procedure is provided that can be invoked by a requesting subject to enable access to certified objects. The *Shurts* approach requires that the requesting subject utilize a particular trusted procedure in order to access otherwise inaccessible objects, and does suggest a file that is configured to define permitted access relationships, as recited by claim 24.

Thus, in view of the above, the combination of *Edwards* and *Shurts* fails to teach or suggest all elements of independent claim 24. Therefore, the rejection of claim 24 should be withdrawn.

Dependent Claims

In view of the above, Applicant respectfully submits that independent claims 1, 12, and 24 are not obvious under 35 U.S.C. § 103 over the combination of *Edwards* and *Shurts*. Further, each of dependent claims 2-11, 13-23, and 25-29 depend either directly or indirectly from one of independent claims 1, 12, and 24, and thus inherit all limitations of the respective independent claim from which they depend. It is respectfully submitted that dependent claims 2-11, 13-23, and 25-29 are allowable not only because of their dependency from their respective independent claims for the reasons discussed above, but also in view of their novel claim features (which both narrow the scope of the particular claims and compel a broader interpretation of the respective base claim from which they depend).

IV. New Claims

New claims 30-33 are added herein. Independent claim 30 recites:

> A method comprising:
> designating sensitivity labels associated with subjects and objects such that each sensitivity label either dominates, is dominated by, or is incomparable to each other sensitivity label;
> defining in a policy file a mapping of allowed communications between subjects and objects associated with different sensitivity labels, wherein said mapping supports mapping with privilege and mapping without privilege; and
> enforcing sensitivity labels by the operating system such that the operating system restricts communications between subjects and objects associated with inconsistent sensitivity labels except as permitted by said mapping. (Emphasis added).

As discussed above with claim 1, the combination of *Edwards* and *Shurts* fails to teach or suggest at least the above-emphasized elements of claim 30. Therefore, claim 30 is believed to be of allowable merit over the applied references.

Each of dependent claims 31-33 depend either directly or indirectly from independent claim 30, and thus inherit all limitations of claim 30. It is respectfully submitted that dependent claims 31-33 are allowable not only because of their dependency from claim 30 for the reasons discussed above, but also in view of their novel claim features (which both narrow the scope of the particular claims and compel a broader interpretation of the respective base claim from which they depend).

V. Conclusion

In view of the above, Applicant believes the pending application is in condition for allowance.

The required fee for this response is enclosed. If any additional fee is due, please charge Deposit Account No. 08-2025, under Order No. 10980679-2 from which the undersigned is authorized to draw.

I hereby certify that this correspondence is being deposited with the United States Postal Service as Express Mail, Label No. EV 568260268US in an envelope addressed to: M/S Amendment, Commissioner for Patents, Alexandria, VA 22313.

Date of Deposit: December 6, 2005

Typed Name: Gail L. Miller

Signature: Gail L. Miller

Respectfully submitted,

By: _____
Jody C. Bishop
Attorney/Agent for Applicant(s)
Reg. No. 44,034
Date: December 6, 2005
Telephone No. (214) 855-8007